Promoting robotic design and entrepreneurship experiences among students and teachers

# Lesson 9:
# Drive Mechanism

**Innovative Technology Experiences for Students and Teachers (ITEST), Professional Development Program, July 2017-19**
Mechatronics, Controls, and Robotics Laboratory, Department of Mechanical and Aerospace Engineering, NYU Tandon School of Engineering

# CONTENTS

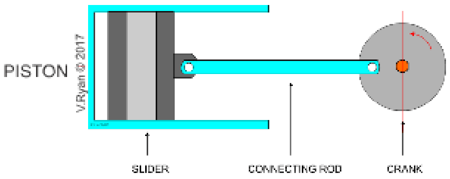

- Types of motion
- Rectangular & polar coordinate systems
- Different types of wheels
- Different types of driving mechanisms
- Theoretical concepts of differential drive

- **TASK/ACTIVITY**: Programming tasks for maneuvering the VEX Claw bot

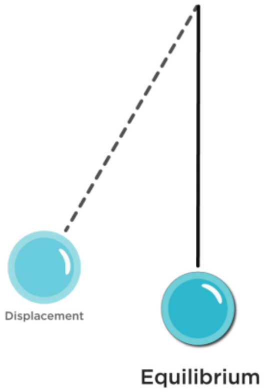
# TYPES OF MOTION

- Linear motion: uniform and non-uniform
- Oscillatory motion
- Periodic motion
- Random motion

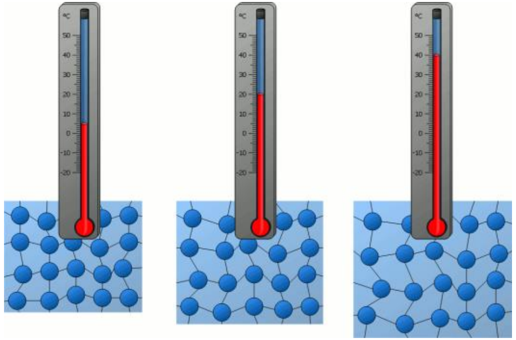
Linear motion




Oscillatory motion



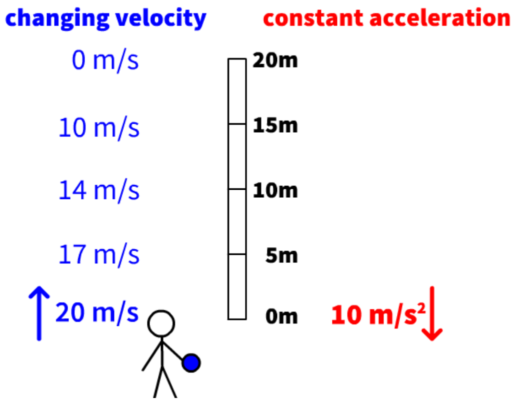

Periodic motion



Random Motion

# LINEAR MOTION

- **Motion along a straight line**
- Examples:
  - Ball thrown straight up and falling back straight down
  - Line following robot on a straight ,etc.
  - Athlete running 100m along a straight track



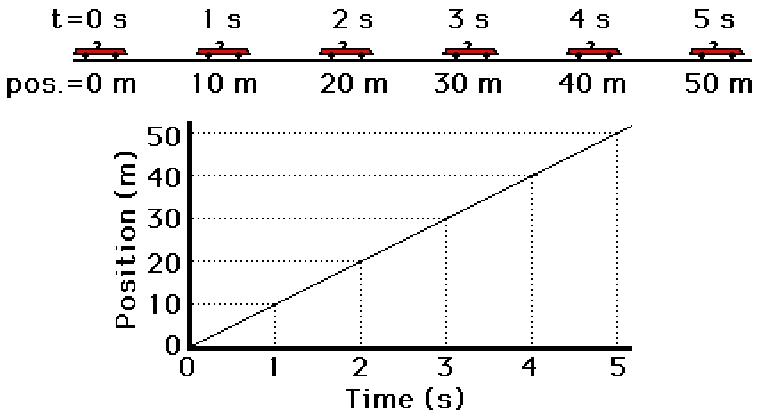

Vertical Linear Motion

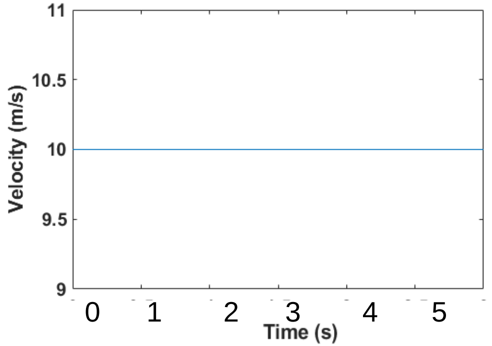
# UNIFORM LINEAR MOTION

- Linear motion in which specific distance is covered in a particular time, is called a uniform linear motion (**speed is constant**)

- Examples:
    - Robot moving on a straight line with constant speed
    - Soldier's marching in a parade at constant speed, i.e., same number of steps per time interval


Source



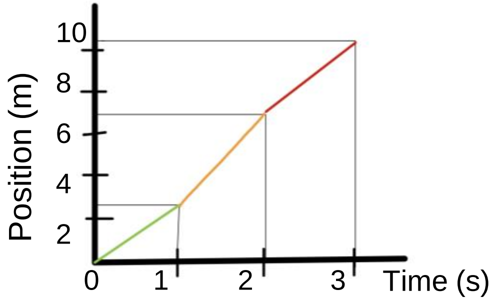



# NON-UNIFORM LINEAR MOTION

- Linear motion that continuously **changes its speed** is called non-uniform linear motion

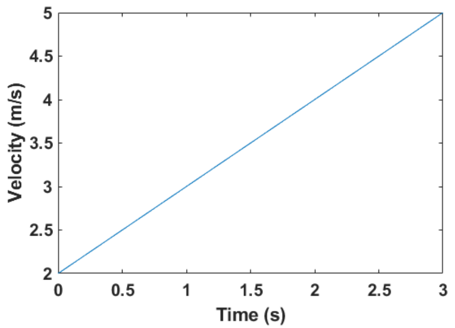
- Examples: Robot accelerating or decelerating, walking with increasing speed (accelerating),


Source




Robot accelerating at 1 $m/s^2$

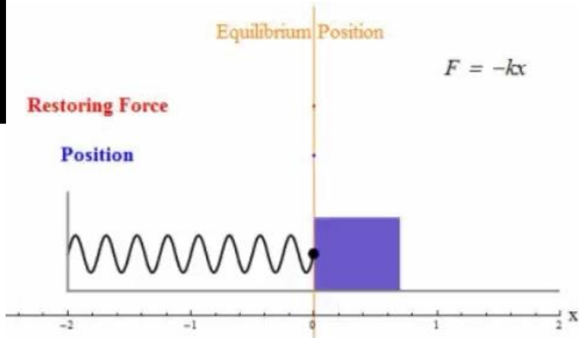
| Time (s) | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| Velocity (m/s) | 2 | 3 | 4 | 5 |



# OSCILLATORY MOTION

- The motion which is back and forth and repetitive is known as oscillatory motion (object repeats the same movement over and over)
- Examples: See-saw, Piston in an engine, mass-spring system, the wings of birds, pendulum, etc.

Piston

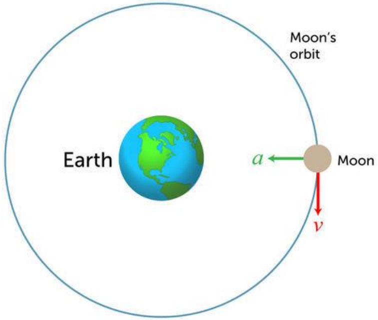




Mass-spring system

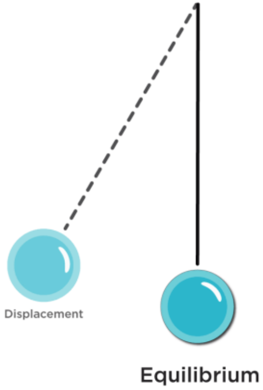
# PERIODIC MOTION

- The motion of moving object that passes through certain point at regular interval of times is called periodic motion
- Examples: Hands of clock, pendulum, orbits of celestial bodies


Source




Source

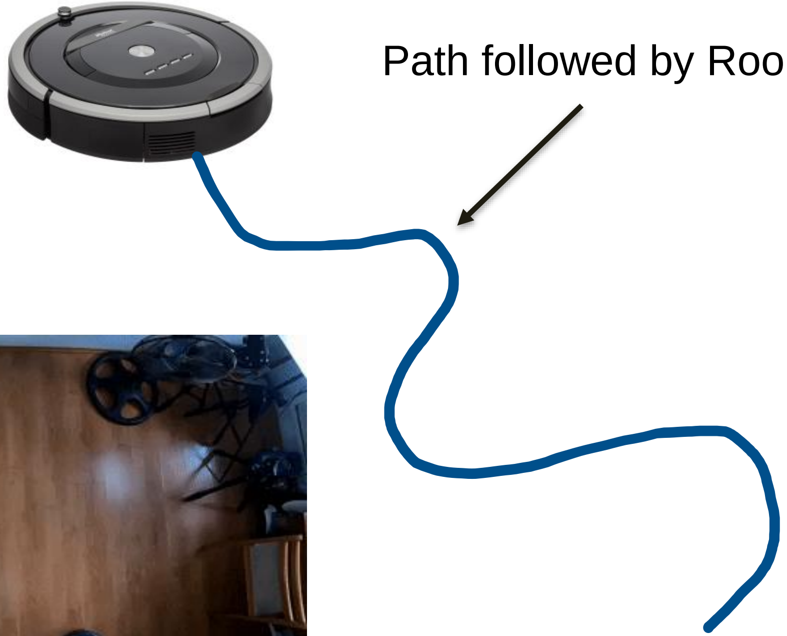



Pendulum

NYU

# RANDOM MOTION

- Motion with continuous changes in direction is called random motion
- Examples: a Roomba cleaning a room, motion of butterfly, honeybees, etc.




Roomba 690 Navigation

# KINEMATICS

- Kinematics is the science of **describing the motion** of objects using words, diagrams, numbers, graphs, and equations

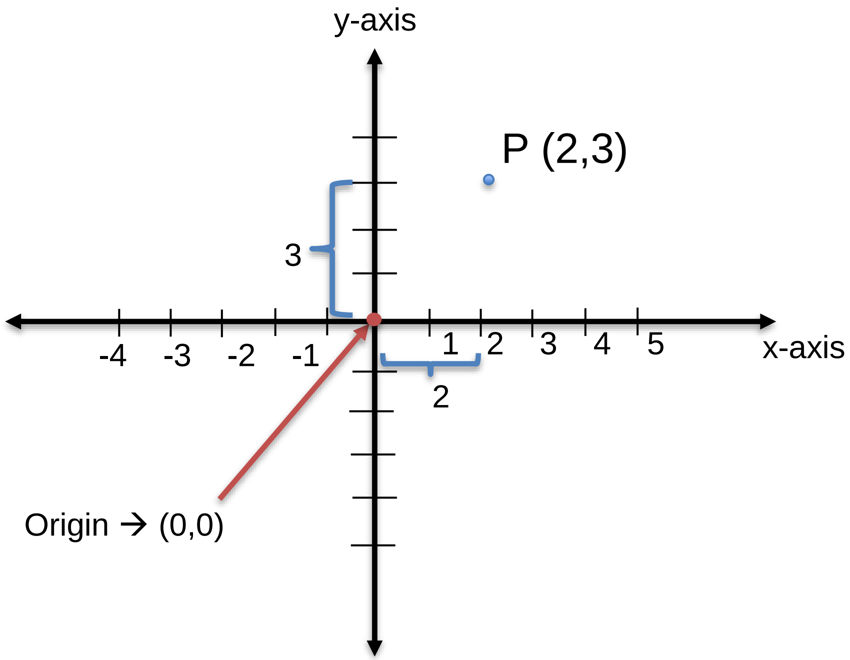
- Motion takes place over time and depends on a frame of reference

- **Frame of reference** – a coordinate system for specifying the precise location of objects in space

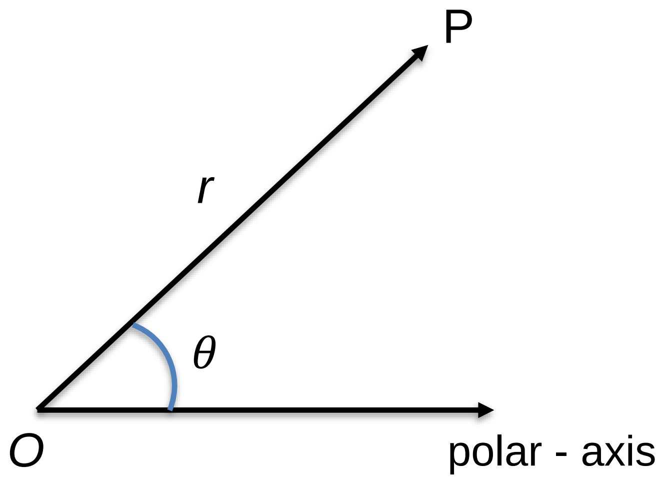
- The **choice of a reference point is arbitrary**, but once chosen, the same point must be used throughout the problem

# COORDINATE SYSTEM

## Cartesian coordinate system




## Polar coordinate system



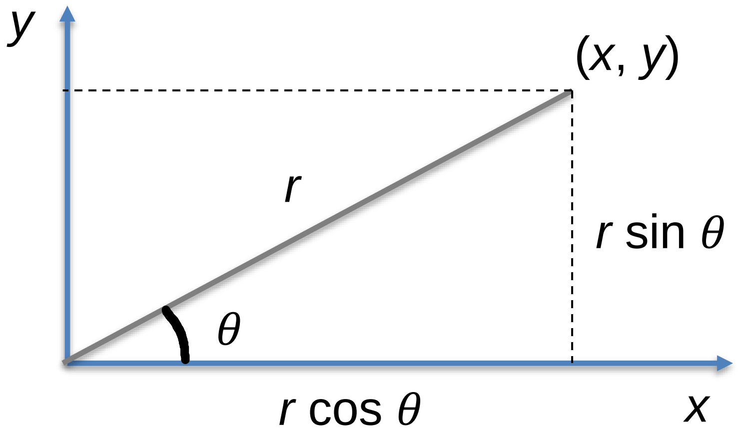
# POLAR COORDINATES TO RECTANGULAR COORDINATES

Given polar coordinates ($r$, $\theta$)

- **Change to rectangular**

By trigonometry

- $x = r \cos \theta$
  $y = r \sin \theta$

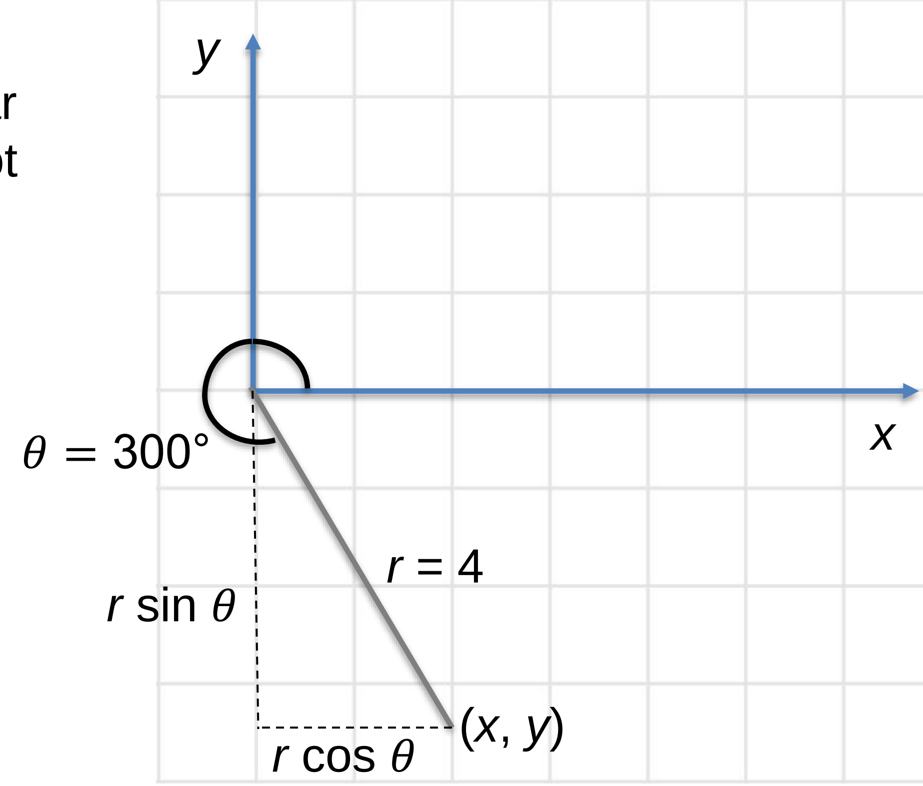



**NOTE**: In Arduino IDE, trigonometric functions take angle input in **radians**

## Problem

If the location of a robot is (4, 300°) in Polar coordinate, what is the location of the robot in Cartesian coordinate?

$$x = r\cos\theta$$
$$y = r\sin\theta$$



NYU

# SOLUTION

## Problem

If the location of a robot is (4, 300°) in Polar coordinate, what is the location of the robot in Cartesian coordinate?

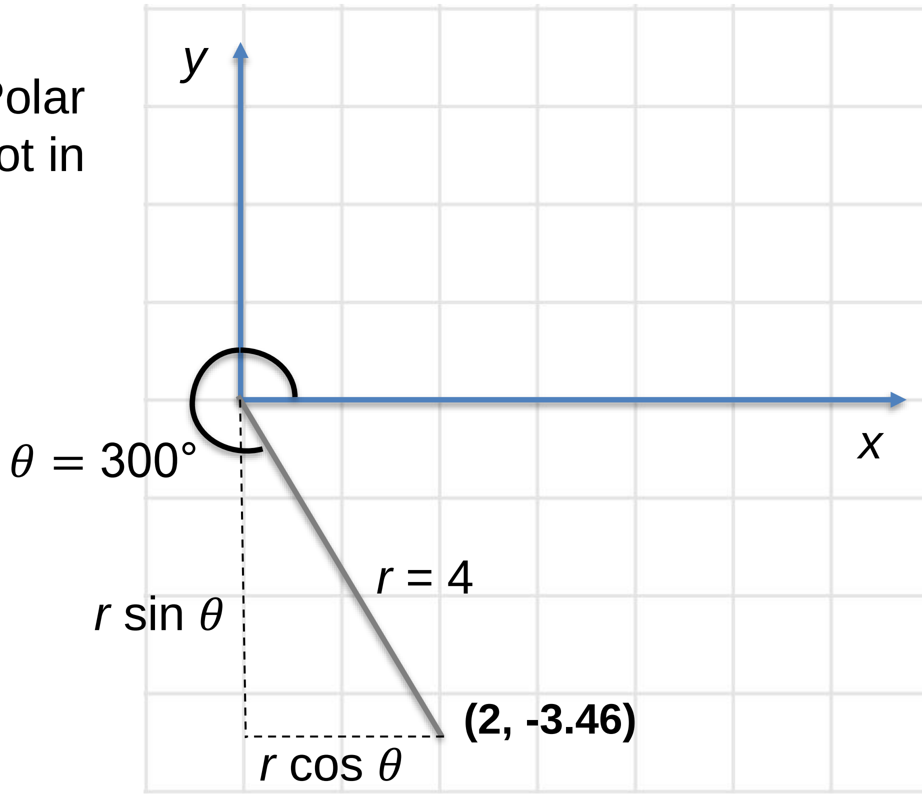
## Solution

Given, polar coordinates: (4, 300°)

Cartesian coordinates:

$x$ = 4 * cos(300°) = **2**

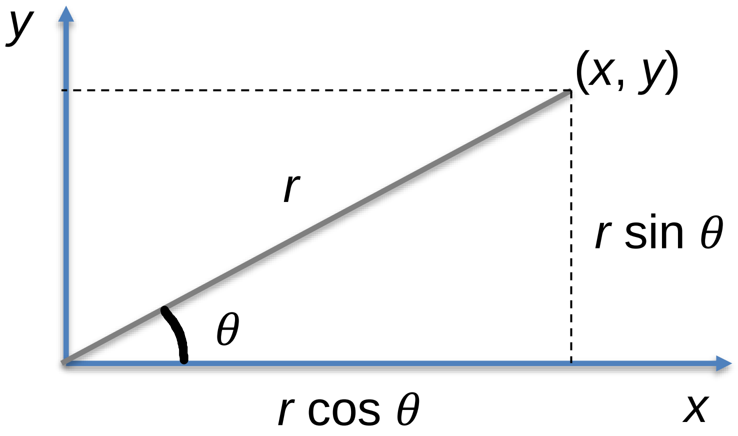
$y$ = 4 * sin(300°) = **-3.46**



# RECTANGULAR COORDINATES TO POLAR COORDINATES

Given a point ($x$, $y$)

- **Convert to ($r$, $\theta$)**




By Pythagorean theorem

$$r^2 = x^2 + y^2$$

By trigonometry

$$\theta = \tan^{-1}\left(\frac{y}{x}\right)$$

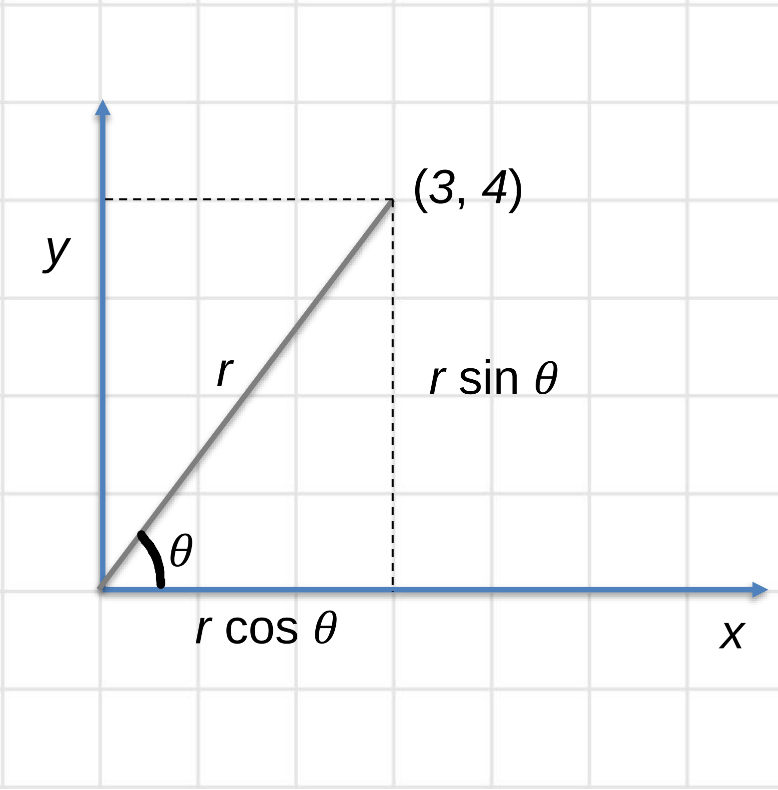
# CONVERTING RECTANGULAR COORDINATES TO POLAR COORDINATES

**Problem**

If the location of a robot is (3, 4) in Cartesian coordinate, what is the location of the robot in Polar coordinate?

$$r^2 = x^2 + y^2$$

$$\theta = \tan^{-1}\left(\frac{y}{x}\right)$$



# SOLUTION

**Problem**

If the location of a robot is (3, 4) in Cartesian coordinate, what is the location of the robot in Polar coordinate?

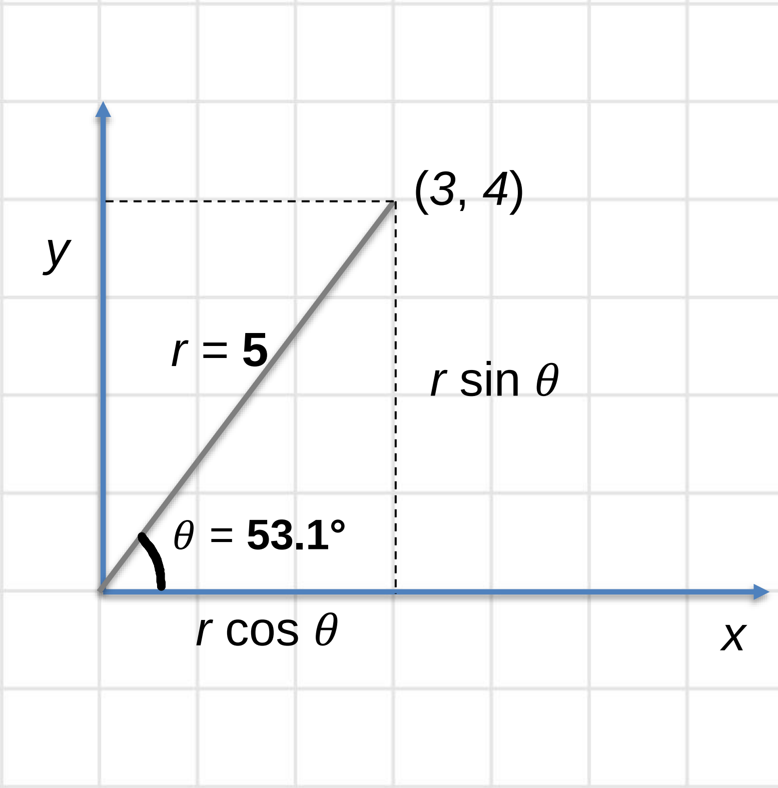
**Solution**

Given, cartesian coordinates: (3, 4)

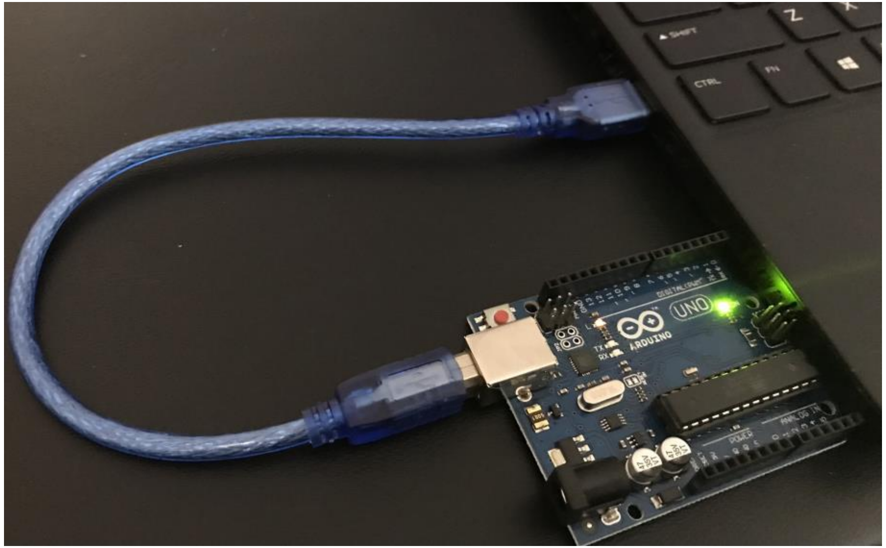
Polar coordinates:

$r = \sqrt{3^2 + 4^2} = \mathbf{5}$

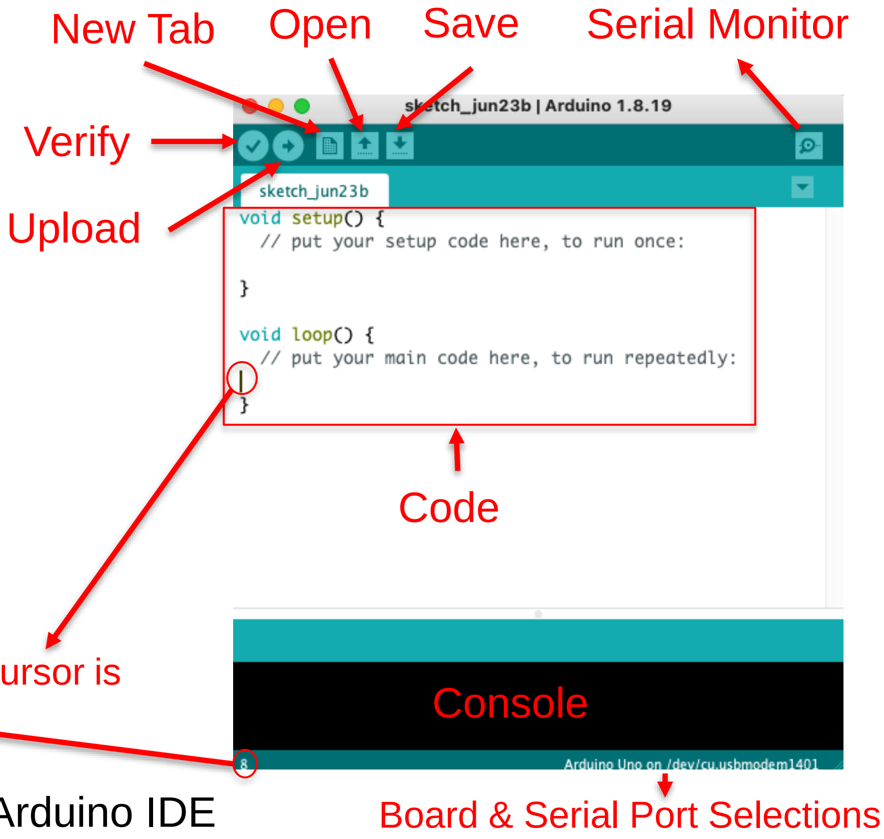
$\theta = \tan^{-1}(\frac{4}{3}) = 0.927 \text{ radians} = \mathbf{53.1°}$



# REFRESHER: ARDUINO




Connect the USB cable




Arduino IDE

# PROGRAMMING

**Serial.begin(rate)**

- Opens serial port and sets the baud rate for serial data transmission
- Typical baud rate for communicating with PC is 9600 although other speeds are supported
- When using serial communication, digital pins 0 (RX) and 1 (TX) cannot be used at the same time

**Serial.print()**

- Prints data to the serial port as human-readable ASCII text.

**String()**

- Constructs a string from the input data (usually of type int/float), which results in a string that contains the ASCII representation of that data

# ACTIVITY 1

**Write a program to display Polar coordinates of robot if Cartesian coordinates of robot location are (5,6)**

# ACTIVITY 1 - SOLUTION

**Write a program to display Polar coordinates of robot if Cartesian coordinates of robot location are (5,6)**

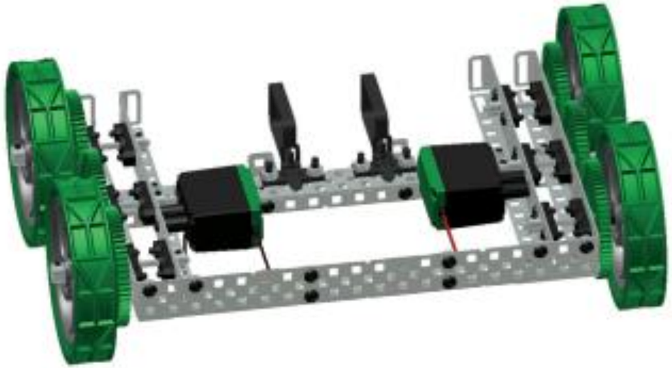
```
float x = 5;
float y = 6;
float r;
float theta;
float pi = 3.14159;

void setup() {
  delay(200);
  Serial.begin(9600);
  /* begin serial communication
  at 9600 baud rate */
  Serial.flush();
  /* Waits for the transmission of
  outgoing serial data to complete */
  r = sqrt(sq(x)+sq(y));
  //  'r' calculation
  theta = atan(y/x);
  // 'theta' calculation
  Serial.print("r = " + String(r, 4) + "\n");
  Serial.print("theta = " + String(theta, 4) + " radians" + "\n");
  Serial.print("theta = " + String((theta*180)/pi)+ " degrees");
  // print statements
}

void loop() {

}
```

```
r = 7.8102
theta = 0.8761 radians
theta = 50.19 degrees
```

Program

# DRIVE TRAIN

- A robot's **drive train** consists of all the components used to make the robot move
  - **Motors**
  - **Wheels**
  - **Transmission**



Source

Source

Source

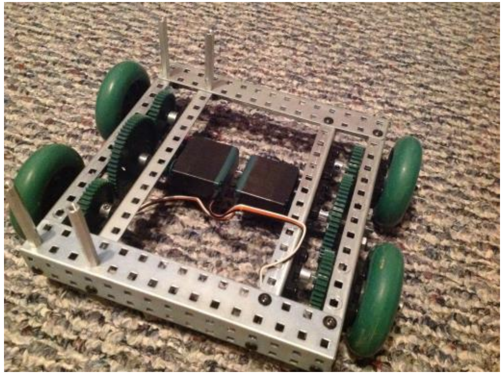
Source

# X-WHEEL DRIVE

**X = number of wheels connected to motors (powered)**

- **Two-**wheel drive: **two** wheels are powered
- **Four-**wheel drive: **four** wheels are powered

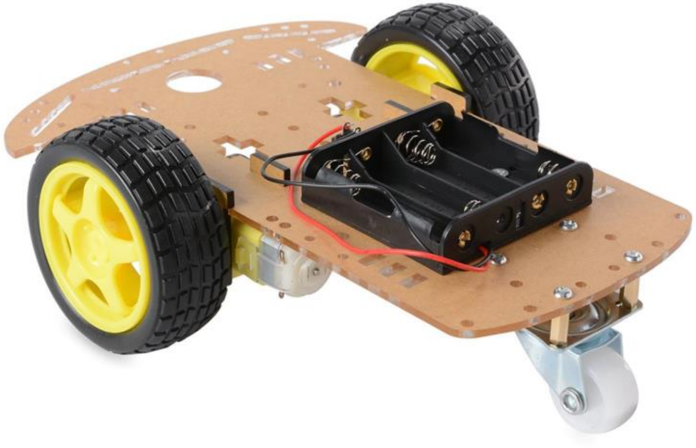
- How many motors the robot has doesn't matter
- For example, the square bot has only two motors, but it is an example of **four-wheel drive** since all four wheels are connected to motors



Source

# CHASSIS

The most common robot chassis are 3 wheeled and 4 wheeled as shown



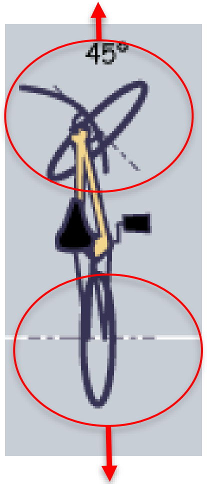
Source

Source

# WHEELS

**Fixed wheel** can rotate about the axis that goes through the center of the wheel and is orthogonal to the wheel plane

**Steerable wheel** has two axes of rotation -- the first is same as the fixed wheel, the second is vertical and goes through the center of the wheel




Source

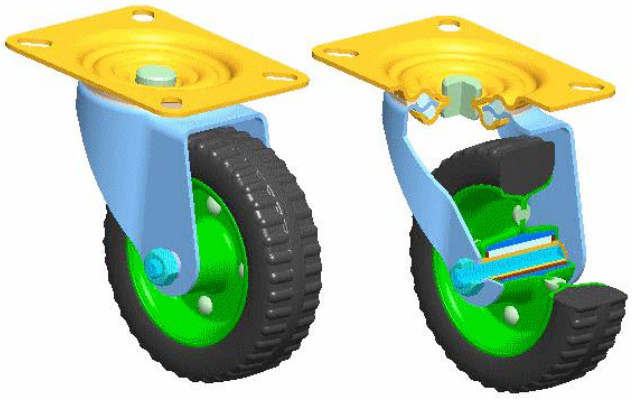
Source

Source

# WHEELS

**Castor wheel** has two axes of rotation, vertical axis does not pass through the center of wheel, from which it is displaced by constant offset

- Such an arrangement causes the wheel to swivel automatically, rapidly aligning with the direction of the motion of the chassis
- This type of wheel is introduced to provide a supporting point for static balance without affecting the mobility of base



Castor Wheel

# WHEELS

**Omni-directional wheel or Swedish wheel** is similar to a normal wheel but has a number of passive rollers around its circumference and their rotational axes lie in the plane of the wheel

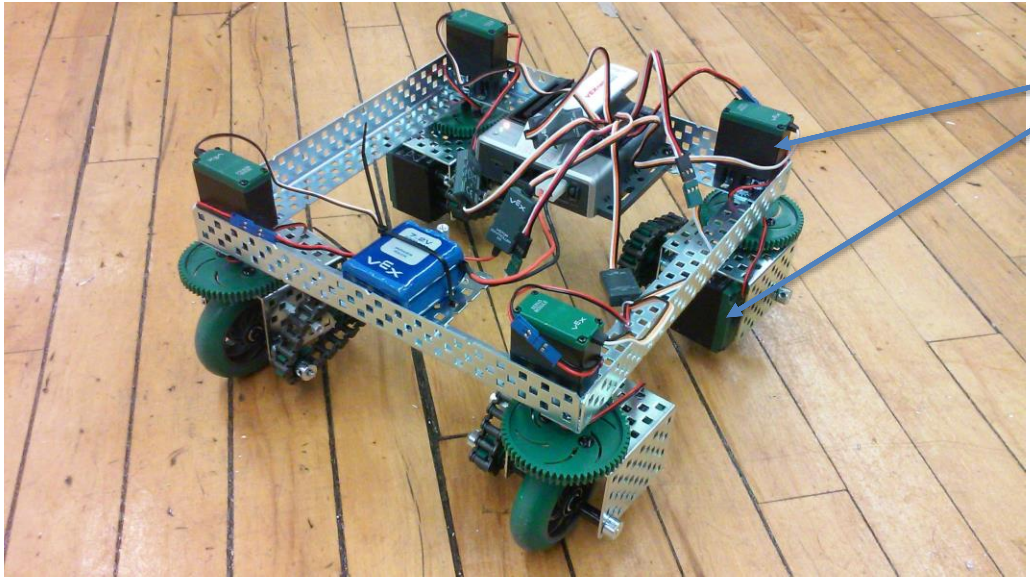
Source


Swedish Wheel

For a **Mecanum wheel,** the axis of rotation of each roller is typically inclined by $45^0$ with respect to the plane of the wheel

Mecanum Wheel

# SWERVE/CRAB DRIVE

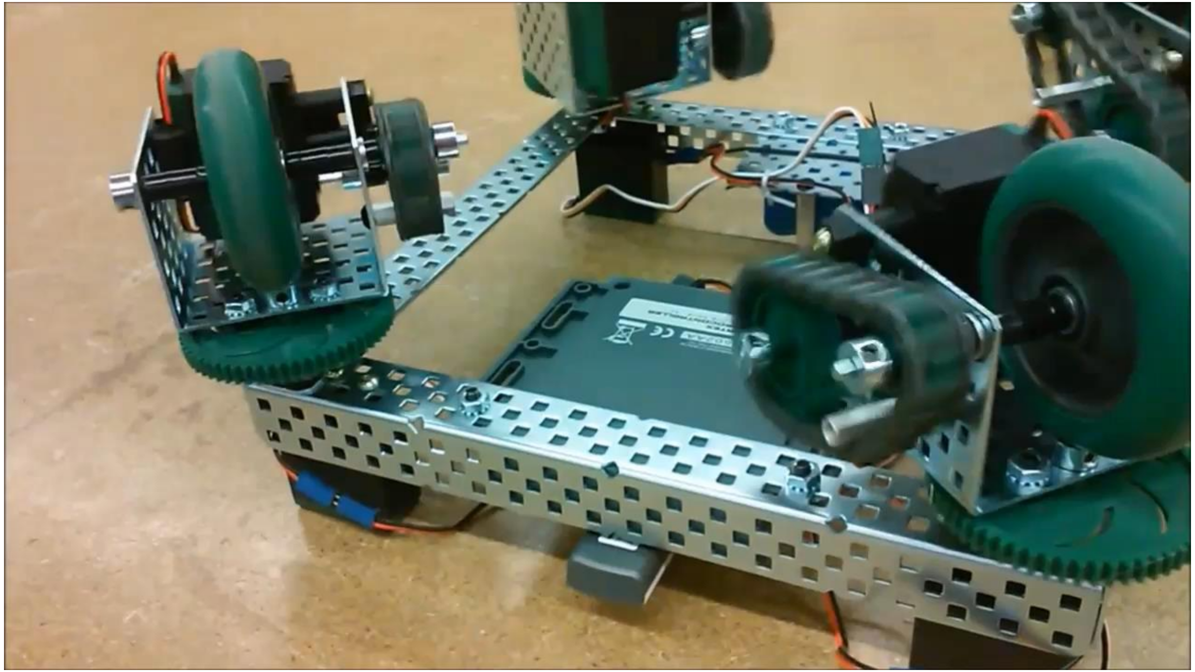


Source

Two motors at each corner

A *swerve drive* robot has the ability to rotate its wheels.

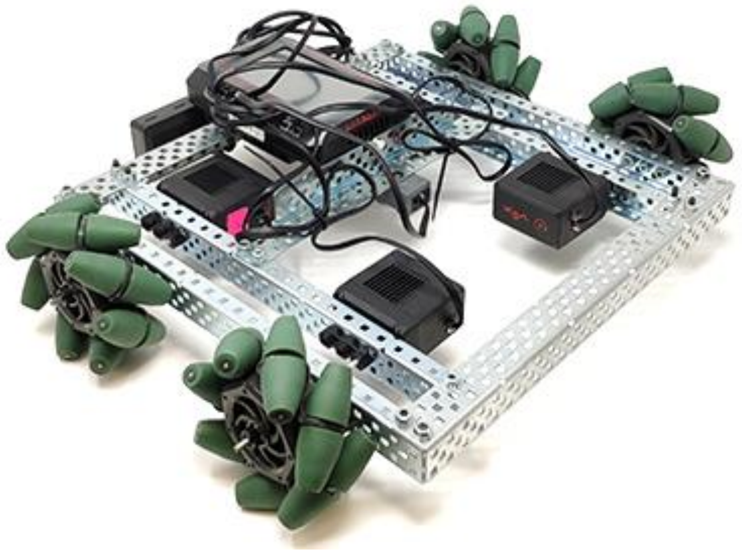
# SWERVE/CRAB DRIVE IN ACTION



[A VEX robot with a swerve drive]

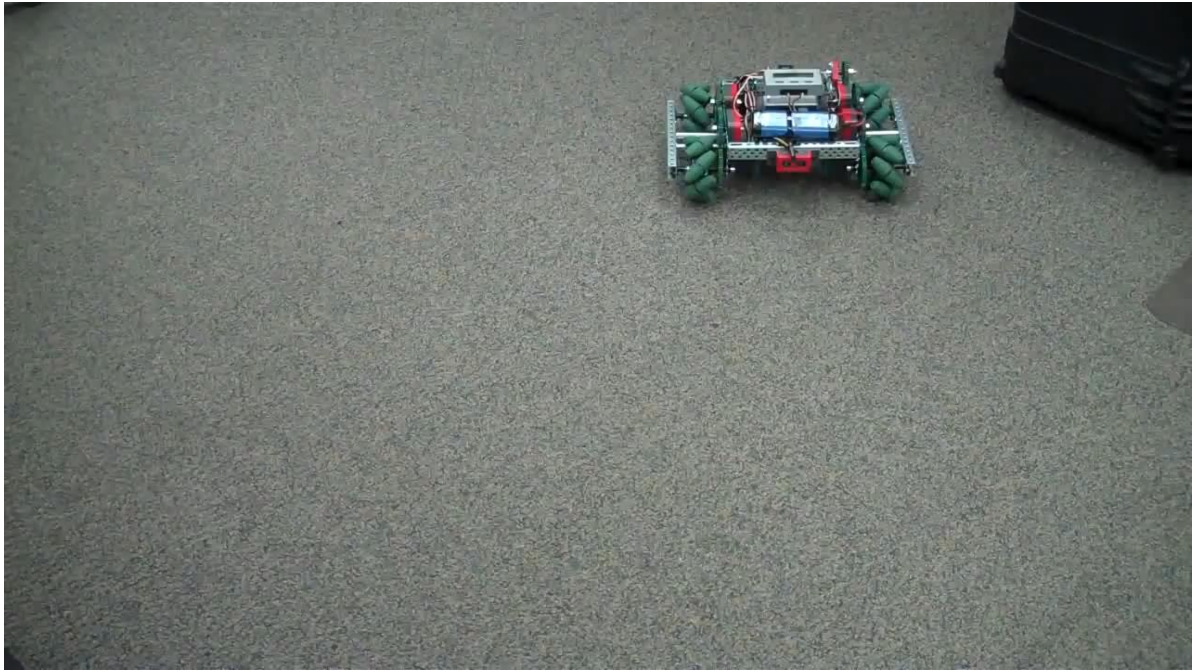
# MECANUM DRIVE

Mecanum drive is an **omnidirectional drive system** where Mecanum wheels are setup in a traditional four-wheel drive system



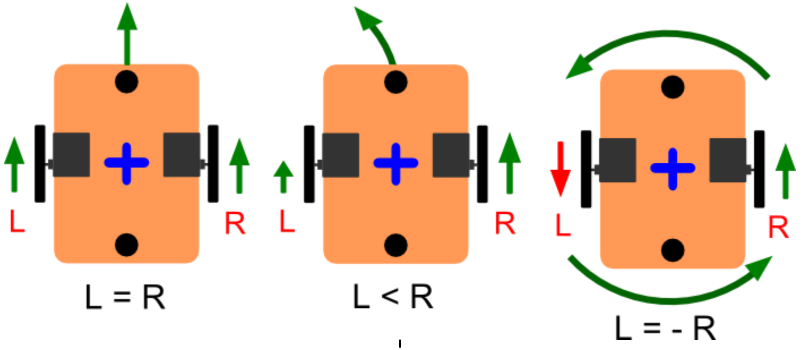
Source

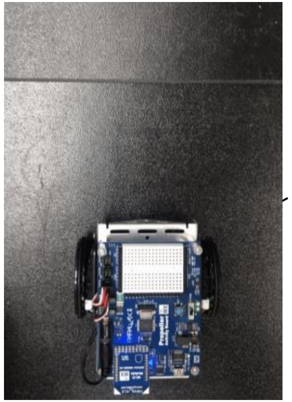
# MECANUM DRIVE IN ACTION



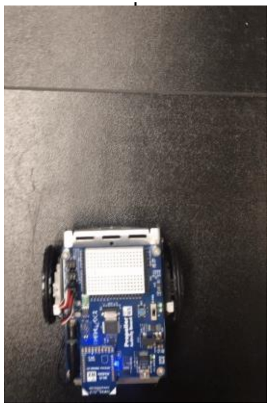
Video: A VEX robot with custom made Mecanum wheels

# DIFFERENTIAL DRIVE KINEMATICS

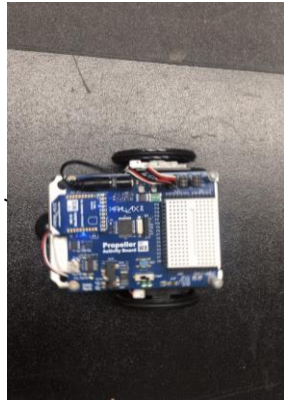


Straight



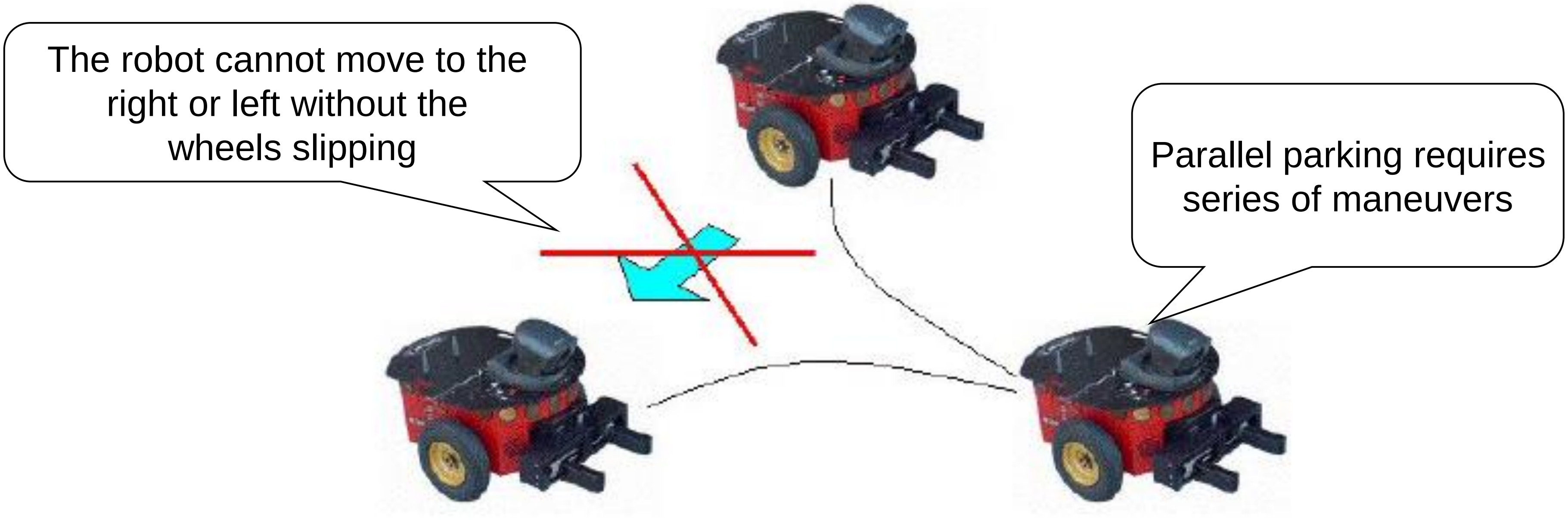
Left turn



Spin

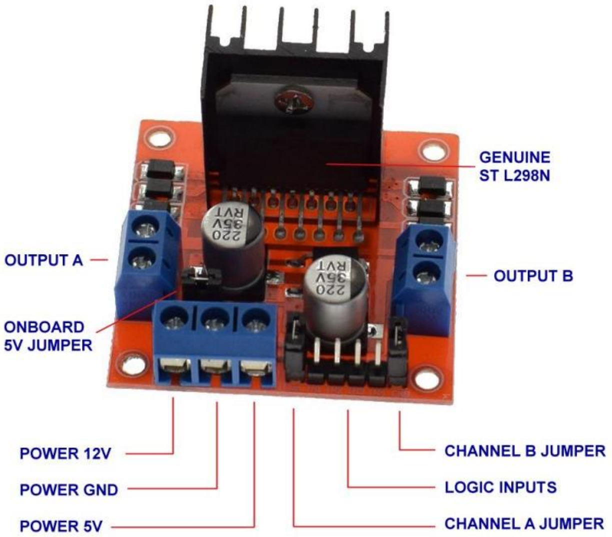
# DIFFERENTIAL DRIVE KINEMATICS

## MAJOR CONSTRAINT




Source

# L298N MOTOR DRIVER



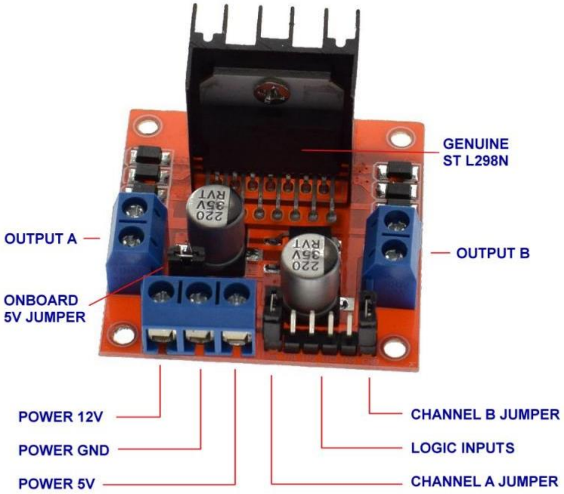

Source

- The L298N is a high current, high voltage **dual full-bridge driver motor** driver IC
- It can control two DC motors simultaneously and independently
- It can provide **2A** per channel at a supply voltage range of **4.5V to 46V**

**NOTE**: L298N is used instead of the L293D since the latter has a limit of only 600mA per channel, and the VEX 393 motor can draw up to 20% of its maximum stall current rating = 0.2 * 4.8A = **0.96A < *2A*** *(output limit of L298N)*

CAUTION: VEX 393 motors draw very high current, do not make circuit changes with the power supply connected, it is dangerous

# L298N MOTOR DRIVER – PINS




Source

| | |
|---|---|
| ENA | Enables PWM signal for Motor A |
| ENB | Enables PWM signal for Motor B |
| IN1 & IN2 | Motor A input pins for direction control of Motor A |
| IN3 & IN4 | Motor B input pins for direction control of Motor B |
| OUT1 & OUT2 | Output pins of Motor A |
| OUT3 & OUT4 | Output pins of Motor B |
| 12 V | 12V input from DC power Source |
| 5 V | Supplies power for the switching logic circuitry inside the IC |
| GND | Ground PIN |

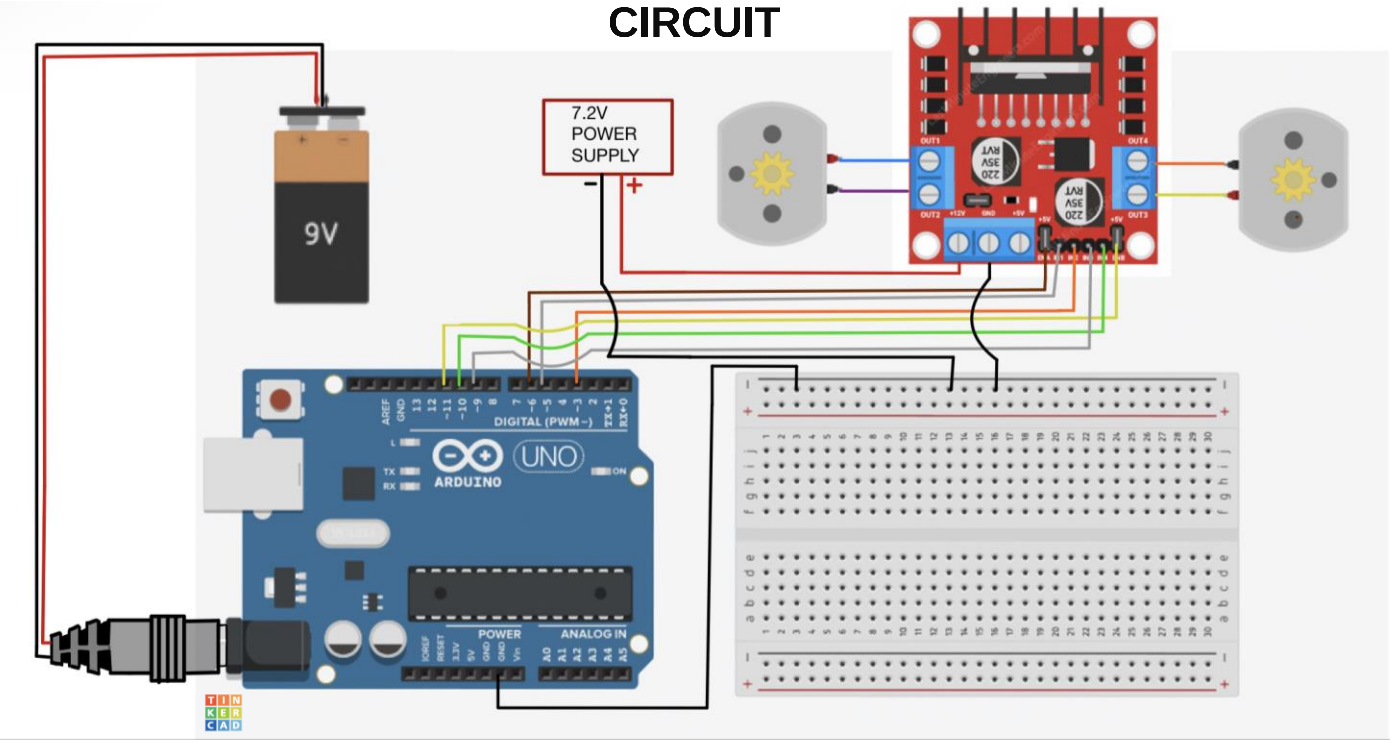
CAUTION: VEX 393 motors draw very high current, do not make circuit changes with the power supply connected, it is dangerous

# ACTIVITY 2

**Make the robot move forward at a specific speed (refer worksheet: Workseet_lesson_8)**

# ACTIVITY 2 - SOLUTION

## CIRCUIT




**NOTE**: For the 7.2V DC power supply, a battery pack of six rechargeable 1.2V batteries can be used

# ACTIVITY 2 - SOLUTION

## CODE

```
int M1IN1 = 3; // right motor input 1
int M1IN2 = 5; // right motor input 2
int EN1 = 6;  // right motor enable pin

int M2IN1 = 9; // left motor input 1
int M2IN2 = 10; // left motor input 2
int EN2 = 11;  // left motor enable pin

void setup(){
  pinMode(M1IN1, OUTPUT);
  pinMode(M1IN2, OUTPUT);
  pinMode(EN1, OUTPUT);
  pinMode(M2IN1, OUTPUT);
  pinMode(M2IN2, OUTPUT);
  pinMode(EN2, OUTPUT);
  /* set IN pins and EN pins in OUTPUT mode for both motors */
}
```

```
void loop() {
    analogWrite(EN1,255); // right motor speed
    analogWrite(EN2,255); // left motor speed

    digitalWrite(M1IN1, HIGH); // right motor CW
    digitalWrite(M1IN2, LOW);

    digitalWrite(M2IN1, LOW); // left motor CCW
    digitalWrite(M2IN2, HIGH);
    delay(1000);

}
```

*Program*

# ACTIVITY 2 - DEMO

Video: Full speed

```
analogWrite(EN1,255);
// right motor speed
analogWrite(EN2,255);
// left motor speed
```

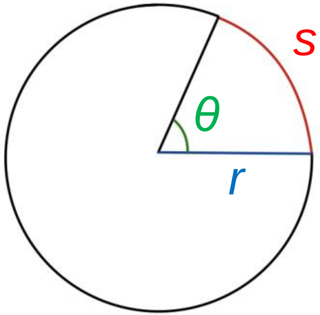
Video: Medium speed

```
analogWrite(EN1,160);
// right motor speed
analogWrite(EN2,160);
// left motor speed
```

Video: Low speed

```
analogWrite(EN1,80);
// right motor speed
analogWrite(EN2,80);
// left motor speed
```

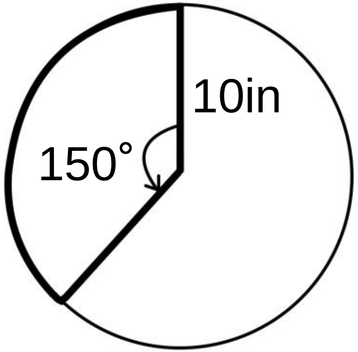
# FINDING ARC LENGTH




$$s = r\,\theta$$

$$\frac{degrees}{180} = \frac{radians}{\pi}$$

$$\pi = 3.14$$



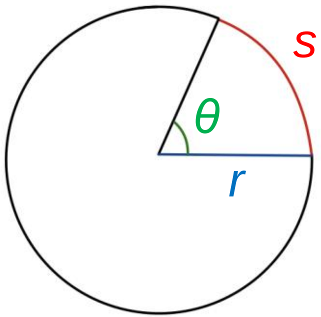

What is the arc length here?

*s* : arc length

*r* : radius of circle

*θ* : angle produced by the arc, measured in radians

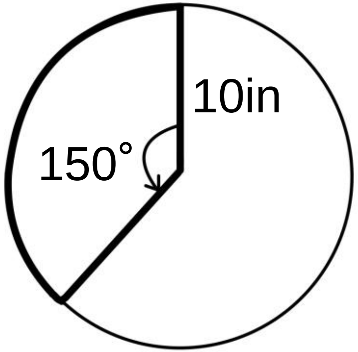
# FINDING ARC LENGTH




$$s = r\,\theta$$

$$\frac{degrees}{180} = \frac{radians}{\pi}$$

$$\pi = 3.14$$




*s* : arc length

*r* : radius of circle

*θ* : angle produced by the arc, measured in radians

What is the arc length here?

*r* = 10in = 10 * 2.54 cm = 25.4cm

*θ* = 150˚ = 150 * (π /180) = 2.62 radians

*s* = *r θ* = 25.4 * 2.62 = 66.55 cm

# KINEMATIC EQUATIONS FOR MOBILE ROBOT (DIFFERENTIAL DRIVE)

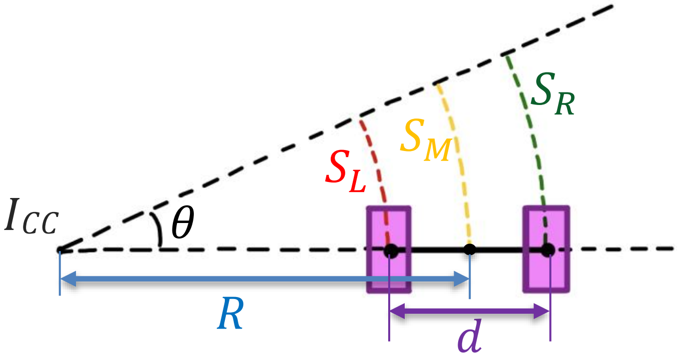
We want the robot to turn from an initial attitude to attitude $\theta$, in time $t$

Distance traveled by **left** (internal) wheel:

$$S_L = \left(R - \frac{d}{2}\right)\theta$$

Distance traveled by **right** (external) wheel:

$$S_R = \left(R + \frac{d}{2}\right)\theta$$




Distance traveled by **mid-point** of wheel-axis:

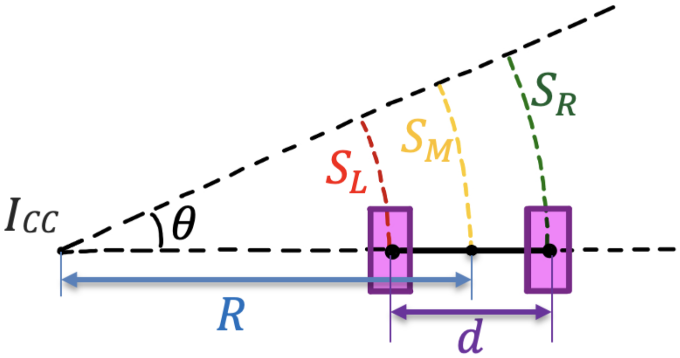
$$S_M = R\,\theta$$

$I_{CC}$ is the instantaneous center of curvature about which the robot rotates

# KINEMATIC EQUATIONS FOR MOBILE ROBOT (DIFFERENTIAL DRIVE)

**Problem**

What is the distance travelled by the left wheel if robot with base length of 6 inches rotates by 30 degrees, given R (distance between center of robot and $I_{CC}$ = 30 inch)?




Distance traveled by **left** (internal) wheel:

$$S_L = \left(R - \frac{d}{2}\right)\theta$$

Distance traveled by **right** (external) wheel:

$$S_R = \left(R + \frac{d}{2}\right)\theta$$

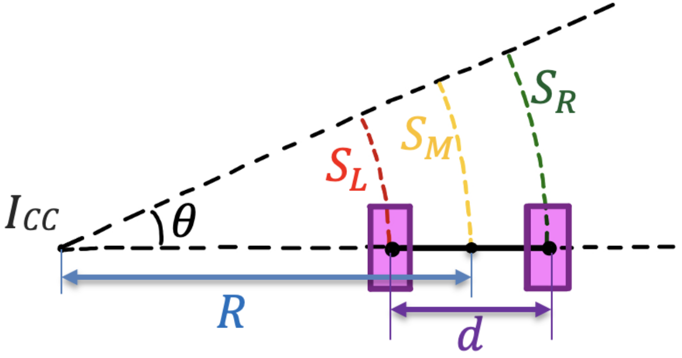
Distance traveled by **mid-point** of wheel-axis:

$$S_M = R\,\theta$$

# KINEMATIC EQUATIONS FOR MOBILE ROBOT (DIFFERENTIAL DRIVE)

**Problem**

What is the distance travelled by the left wheel if robot with base length of 6 inches rotates by 30 degrees, given R (distance between center of robot and $I_{CC}$ = 30 inch)?




**Solution**

Distance traveled by **left** (internal) wheel:

$$S_L = \left(R - \frac{d}{2}\right)\theta$$

$= \left(30 - \frac{6}{2}\right)\left(30 * \frac{\pi}{180}\right) = 27 * 0.524$ = **14.12 inches**

# KINEMATIC EQUATIONS FOR MOBILE ROBOT (DIFFERENTIAL DRIVE)

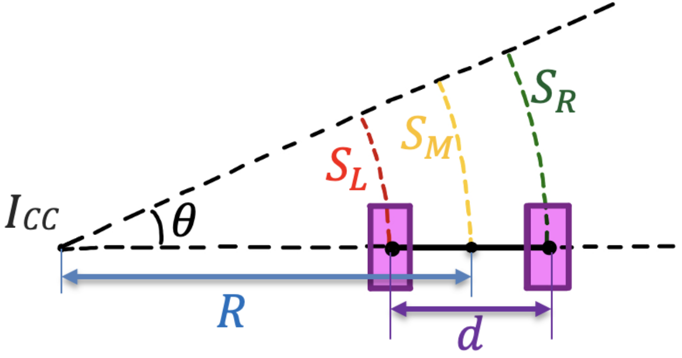
We want the robot to turn from an initial attitude to attitude $\theta$, in time $t$

Distance traveled by **left** (internal) wheel:

$$S_L = \left(R - \frac{d}{2}\right)\theta$$

Distance traveled by **right** (external) wheel:

$$S_R = \left(R + \frac{d}{2}\right)\theta$$




Distance traveled by **mid-point** of wheel-axis:

$$S_M = R\,\theta$$

$I_{CC}$ is the instantaneous center of curvature about which the robot rotates

# KINEMATIC EQUATIONS FOR MOBILE ROBOT (DIFFERENTIAL DRIVE)

We want the robot to turn from an initial attitude to attitude $\theta$, in time $t$

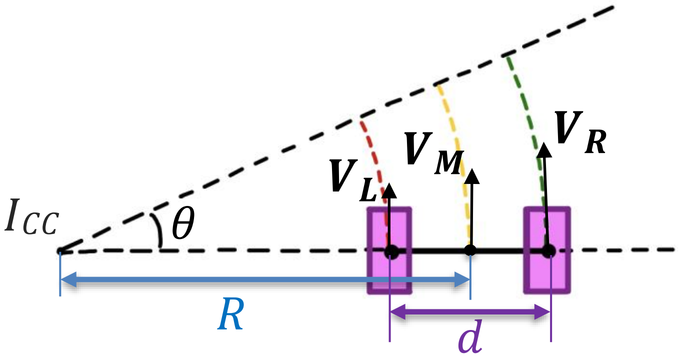
Constant turning rate, $\omega = \frac{\theta}{t}$

Wheel velocities (magnitudes):

$$V_L = \frac{S_R}{t} = \left(R - \frac{d}{2}\right)\frac{\theta}{t} = \left(R - \frac{d}{2}\right)\omega$$

$$V_R = \frac{S_L}{t} = \left(R + \frac{d}{2}\right)\frac{\theta}{t} = \left(R + \frac{d}{2}\right)\omega$$

$$V_M = \frac{S_M}{t} = R\left(\frac{\theta}{t}\right) = R\,\omega$$




$V_L$ & $V_R$ can be calculated if the angle, time to complete the run, $R$ and $d$ are known

Assumptions: Inertia, friction, etc., are negligible
$V_L$ & $V_R$ achieved instantaneously, i.e., no need to worry about accelerating to $V_L$ & $V_R$

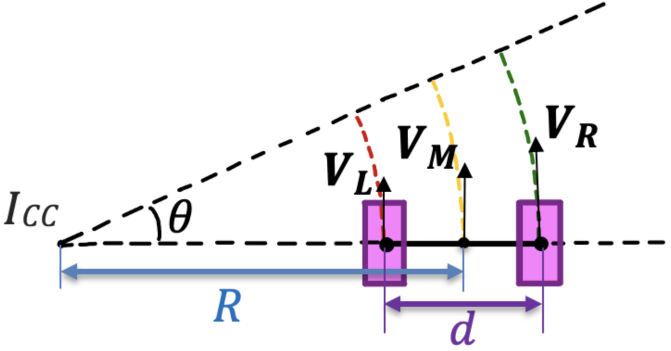
# KINEMATIC EQUATIONS FOR MOBILE ROBOT (DIFFERENTIAL DRIVE)

$V_R = \left(R + \frac{d}{2}\right)\omega$ and $V_L = \left(R - \frac{d}{2}\right)\omega$

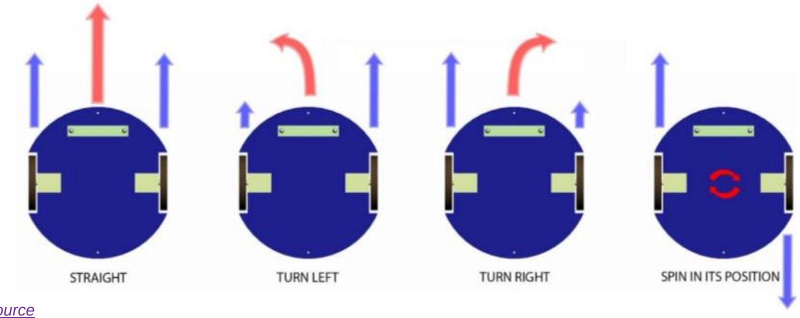
$$V_R + V_L = \left(R + \frac{d}{2} + R - \frac{d}{2}\right)\omega = (2R)\,\omega$$

$$V_R - V_L = \left(R + \frac{d}{2} - R + \frac{d}{2}\right)\omega = d\,\omega \quad \Rightarrow \omega = \frac{V_R - V_L}{d}$$

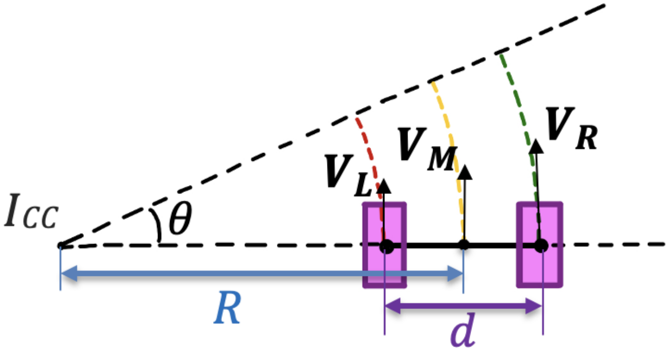
$$\frac{V_R + V_L}{V_R - V_L} = \frac{2R\omega}{d} = \frac{2R}{d} \quad \Rightarrow R = \frac{d(V_R + V_L)}{2(V_R - V_L)}$$



# KINEMATIC EQUATIONS FOR MOBILE ROBOT (DIFFERENTIAL DRIVE)




Source




| | Straight | Turn Left | Turn Right | Spin |
|---|---|---|---|---|
| $R$ | ∞ | $d/2$ (when $V_L = 0$) | $d/2$ (when $V_R = 0$) | 0 |
| $\omega$ | 0 ($V_R = V_L$) | > 0 | < 0 | 0 ($V_R = -V_L$) |

$$\omega = \frac{V_R - V_L}{d}$$

$$R = \frac{d(V_R + V_L)}{2(V_R - V_L)}$$

# ACTIVITY 3

**Make the robot take a 90-degree turn**
**(refer worksheet: Workseet_lesson_8)**

# ACTIVITY 3

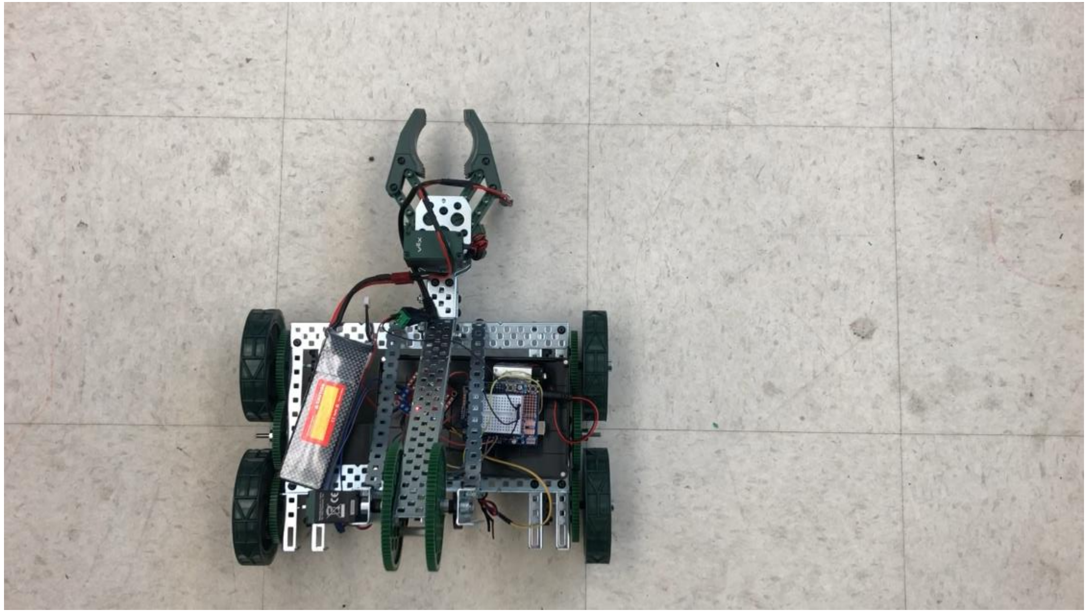
## CODE

```
// right motor
int M1IN1 = 3; // motor 1 input 1
int M1IN2 = 5; // motor 1 input 2
int EN1 = 6;  // motor 1 enable pin

// left motor
int M2IN1 = 9; // motor 2 input 1
int M2IN2 = 10; // motor 2input 2
int EN2 = 11;  // motor 2 enable pin

void setup(){
  pinMode(M1IN1, OUTPUT);
  pinMode(M1IN2, OUTPUT);
  pinMode(EN1, OUTPUT);
  pinMode(M2IN1, OUTPUT);
  pinMode(M2IN2, OUTPUT);
  pinMode(EN2, OUTPUT);
  // set both IN pins and EN pins in OUTPUT mode for both motors
  // set enable pins on L298N HIGH

  analogWrite(EN1,0); // right motor speed
  analogWrite(EN2,255); // left motor speed

  delay(2000); // delay before the bot starts to move

  digitalWrite(M1IN1, HIGH); // right motor CW
  digitalWrite(M1IN2, LOW);

  digitalWrite(M2IN1, LOW); // left motor CCW
  digitalWrite(M2IN2, HIGH);

  delay(1600); /* delay value for 90degrees by trial & error */

  analogWrite(EN2,0); // left motor speed
}


void loop(){

}
```

Program

# ACTIVITY 3 - DEMO

**Note**: Circuit same as activity-2



90-degree turn

# Task / Activity:
# Programming tasks

# TASK / ACTIVITY

- Experiments with differential drive mechanisms
- Moving forward, backward, turning left or right (for turning changing speed/velocity and direction)

# Thank You!

## Questions and Feedback?